DOI:
10.1109/TNSRE.2015.2507941

Document Version
Peer reviewed version

Learning Predictive Movement Models from Fabric-mounted Wearable Sensors

Brendan Michael and Matthew Howard

***Abstract*—The measurement and analysis of human movement for applications in clinical diagnostics or rehabilitation is often performed in a laboratory setting using static motion capture devices. A growing interest in analysing movement in everyday environments (such as the home), has prompted the development of "wearable sensors", with the most current wearable sensors being those embedded into clothing. A major issue however with the use of these fabric-embedded sensors is the undesired effect of fabric motion artefacts corrupting movement signals. In this paper, a non-parametric method is presented for learning body movements, viewing the undesired motion as stochastic perturbations to the sensed motion, and using orthogonal regression techniques to form predictive models of the wearer's motion that eliminate these errors in the learning process. Experiments in this paper show that standard non-parametric learning techniques under-perform in this fabric motion context, and that improved prediction accuracy can be made by using orthogonal regression techniques. Modelling this motion artefact problem as a stochastic learning problem shows an average 77% decrease in prediction error in a body pose task using fabric-embedded sensors, compared to a kinematic model.**

I. Introduction

Measuring and analysing human movement is important in a wide range of fields such as analysing human gait for rehabilitation [1], medical diagnostics [2], or even animation control [3]. In the laboratory or studio environment, a variety of devices are available for taking such measurements, including sophisticated inertial motion capture systems [4] or other strap-on sensors [5].

However, even wearable sensors designed for use outside the laboratory environment [6] [7], can present a number of practical problems. These include (i) size and weight of sensors affecting user movements (undesirable for analysis/diagnostic studies), (ii) sensors being visually obtrusive (detrimental to patient experience [8]), and (iii) inability to record over a continuous time-period lasting days or even weeks.

A natural solution to these problems is to incorporate such systems into items already in use by patients. The development of *fabric-embedded sensors*, also known as *e-textiles*, is an emerging technology that uses small sensors mounted onto items of fabric, such as clothing [9]. This can include sensors for measuring body kinematics, such as the ITG3205 gyroscope (see Fig. 1), or the ADXL335 accelerometer (see Sec. IV-B, and Fig. 8). These sensors are very small in size (order of millimetres in diameter) and are built in mind of the user's comfort by minimising physical and visual invasiveness. In comparison to laboratory devices, they are also very inexpensive.

While these new, soft sensing technologies offer significant potential for inexpensive and unobtrusive capture of human movement data, there remain a number of problems in their use. An important issue, is that of how to deal with motion artefacts corrupting data recordings, as caused by the unpredictable motion of fabric sensors with respect to the body (see Fig. 1). Previous fabric systems have attempted to solve this problem by fitting sensors tightly to the body (e.g., by use of straps or other tight-fitting garments [9]). However, this is unsatisfactory if sensors are to be incorporated in an unobtrusive way, into everyday items of clothing.

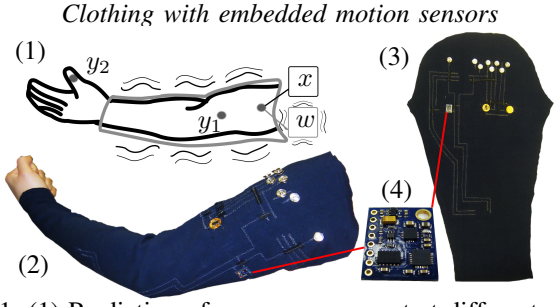

Fig. 1: (1) Prediction of wearer movement at different points on the body (e.g., forearm y_1, fingertip y_2) based on sensor readings. Fabric motion with respect to the body introduces unpredictable artefacts into e-textile sensor readings w as compared to those derived from a rigidly attached sensor x. (2) Sleeve with embedded sensors. (3) Disassembled sleeve showing connections made with conductive thread. (4) Inertial measurement unit embedded on fabric.

As an alternative, this paper proposes the use of statistical methods from the errors-in-variables field to learn models of the wearer's movement that eliminate the effects of fabric motion artefacts. The proposed approach is computationally efficient, and can be easily implemented in an embedded system for on-board (i.e., on-wearer) prediction of movements. In this paper, experimental results in learning and predicting movement from acceleration data from a physical e-textile device indicate superior performance as compared to standard learning approaches.

II. Problem Definition

To motivate the proposed approach, consider the effect that non-rigid attachment of body-mounted sensors may have on estimation of a wearer's movement (see Fig. 1). In a typical problem, one might have access to readings w from a sensor (e.g., accelerometer) mounted on an item of clothing (e.g., shirt sleeve). From these, it is desired to estimate the corresponding motion (accelerations) of different points on the wearer's body. For example, these might include estimating the acceleration of the forearm y_1, or a fingertip y_2.

Given readings from a sensor located rigidly on the arm x, this can be achieved in a straight forward way, either through forward kinematics (given an analytic model of the wearer's

arm), or by learning the mapping from x to y_1 or y_2 from a set of calibration data.

However, in the case of sensing from e-textiles, the loose coupling between the wearer motion and sensor readings, means that artefacts are introduced from a number of sources, for example, external air currents, air resistance on the fabric, vibrations and the dynamics of the fabric itself. These unpredictable disturbances may cause a significant discrepancy between the sensor readings w and the underlying motion of the wearer.

A. Explicit Models of Fabric Dynamics

One approach to deal with these issues would be to explicitly model the wearer/ fabric interaction dynamics. For example, in the field of cloth animation, fabric dynamics have been simulated using mass-spring models or particle systems based on motion capture data [10]. This, however, is a complex procedure, due to the need to first analyse the mechanics of interactions, then build a suitable model to represent these interactions [11]. The computational demands of predicting on such a model, make them unsuitable for a light-weight, embedded wearable system. Furthermore, different materials may interact in different ways, depending on their internal fabric, or fibre structure [12]. This means that models formed in this way are restricted to a particular class of fabric, and become prohibitively complex with the addition of differing materials.

B. Learning Fabric Motions

In the absence of a detailed model of the wearer/fabric interaction dynamics, an appealing approach is to use statistical learning techniques to form a predictive model of the wearer's motion. Addressing the problem in this way provides many benefits over analytical fabric modelling, since it allows unpredictable motion artefacts to be treated as stochastic perturbations to the underlying motion. It removes the need to estimate physical quantities such as mass and fibre structure of the fabric, and through use of simple parametric models, can be computationally very inexpensive.

To apply such an approach, a calibration stage is required in which data from the the target quantity $y \in \mathbb{R}$ and the fabric sensor readings $\mathbf{w} \in \mathbb{R}^{\mathcal{P}}$ are gathered for training the model. In the setting considered here (ref. Fig. 1), such data may be gathered by subjecting the system to various movements while data is recorded both from the fabric sensors and from a sensor measuring the target quantity.

Note that, since the latter is only needed temporarily (i.e., during the calibration), a larger, rigidly-attached sensor can be used, that may otherwise not be suitable for long-term use. For example, one might chose to use a more intrusive, but higher fidelity motion capture sensor to obtain high quality readings, knowing that once the calibration is complete, the rigid sensor may be discarded, in favour of the predictions obtained from the fabric sensor readings.

C. Standard Least Squares Estimators

While the approach described above is appealing for dealing with fabric-mounted sensor data, close examination of the usual assumptions underlying standard learning suggest its direct application may be problematic. This is due to an important mismatch in the sources of error expected by these approaches, and those actually encountered in the data.

Specifically, the standard assumption made by such techniques is that data are generated according to a model of the form

$$y = f(\mathbf{x}) + \epsilon \tag{1}$$

where ϵ denotes additive noise on y and f denotes the functional relationship between the sensed inputs $\mathbf{x} \in \mathbb{R}^{\mathcal{P}}$ and the target outputs y. Given samples $\{\mathbf{x}_n, y_n\}_{n=1}^{\mathcal{N}}$ the goal of the learning approach is to form an estimate of the function f.

A common approach for this is to minimise the sum of the squared residuals

$$S_o = \sum_{n=1}^{\mathcal{N}} (y_n - \tilde{y}_n)^2 \tag{2}$$

where $\tilde{y}_n$ denotes the prediction of the model on the nth data point. A convenient class of function approximators are the generalised linear models

$$\tilde{y} = \boldsymbol{\phi}(\mathbf{x})^\top \tilde{\boldsymbol{\theta}} \tag{3}$$

where $\boldsymbol{\phi}(\cdot) \in \mathbb{R}^{\mathcal{J}}$ is a suitable feature vector or set of basis functions, such as Gaussian radial basis functions or polynomials and $\tilde{\boldsymbol{\theta}} \in \mathbb{R}^{\mathcal{J}}$ is the parameter. Note that, for convenience, in this paper it is assumed that $\boldsymbol{\phi}(\cdot)$ contains the term $\phi_{\mathcal{J}}(\cdot) := 1$ to encode any constant offset in the target function (1).

The optimal choice for the parameter (with respect to (2)) is

$$\tilde{\boldsymbol{\theta}} = (\boldsymbol{\Phi}^\top \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^\top \mathbf{y} \tag{4}$$

where $\boldsymbol{\Phi} := (\boldsymbol{\phi}_1^\top, \cdots, \boldsymbol{\phi}_{\mathcal{N}}^\top)^\top \in \mathbb{R}^{\mathcal{N} \times \mathcal{J}}$ is the data matrix, containing independent sample features $\boldsymbol{\phi}_n := \boldsymbol{\phi}(\mathbf{x}_n)$ on each column.

An alternative to learning a global function is to fit spatially localised low order polynomials [13] (often linear or quadratic [14]) in the original input space. This nonparametric estimation method allows for improved scalability in terms of the dimensionality of the data, does not require biases on the data to be specified [15] (e.g. the parametric form of the data), and also avoids the problems of global interference [16]. A weighting function is used to determine a sample's contribution λ_n to the parameter estimation of a model, generally based on the input's distance from the centre of the region $\mathbf{c}_i \in \mathbb{R}^{\mathcal{P}}$.

As such, instead of minimising the sum of squared residuals (2), the objective function for local learning minimises

$$S_{ow} = \sum_{n=1}^{\mathcal{N}} \lambda_n (y_n - \tilde{y}_n)^2. \tag{5}$$

For each model, the diagonal weight matrix $\boldsymbol{\Lambda} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ is

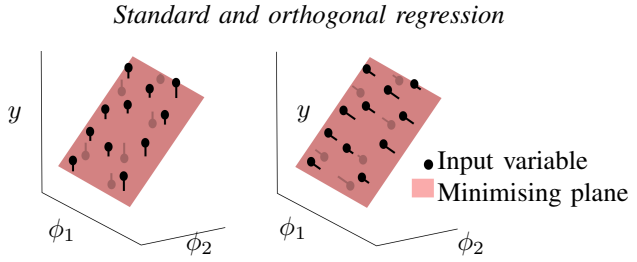

Fig. 2: Fitting the model (1) with standard least squares (left) minimises the residuals due to error in the target variable y but ignores errors in the inputs. By minimising the residuals orthogonal to the fit, total least squares (right) reduces their effect.

formed for all samples, and the weighted least squares solution for the ith model becomes:

$$\tilde{\boldsymbol{\theta}}_i = (\boldsymbol{\Phi}^\top \boldsymbol{\Lambda}_i \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^\top \boldsymbol{\Lambda}_i \mathbf{y} \tag{6}$$

In the case where $\boldsymbol{\Lambda} = \mathbf{I}$, this is equivalent to the least squares formulation (4).

In the context of fabric-based sensing, difficulties occur due to (1) being a poor model of the noise encountered in the data. As described in §II-B, the major source of noise in fabric sensor data is that of the motion of fabric with respect to the wearer, i.e., *noise on the inputs* $\mathbf{x}$. This has a number of implications with respect to the reliability of movement prediction models computed according to this standard approach. For example, ignoring these perturbations, and using the least squares estimate ((4),(6)) may result in (i) bias in the parameter estimation (ii) loss of power in detecting relationships, and (iii) the masking of features in non-linear models [17].

III. Method

To deal with these problems, in this paper it is proposed to explicitly account for non-negligible noise in the *independent variables* $\mathbf{x}$ during learning and prediction. Specifically, the latter are assumed to be subject to zero-mean, additive noise $\boldsymbol{\varepsilon}$, that corrupt the sensor readings

$$\mathbf{w} = \mathbf{x} + \boldsymbol{\varepsilon}. \tag{7}$$

Given samples $\{\mathbf{w}_n, y_n\}_{n=1}^{\mathcal{N}}$, the task is to form a prediction model (3), that takes account of stochasticity both in $\mathbf{x}$ and y in the data.

A. Model Estimation through Total Least Squares

An intuitive approach to achieve this, is to modify the objective function (2) such that the parameter estimate minimises the squared residuals orthogonal to the predicted curve, an approach known as Total Least Squares (TLS) fitting [18], [19]. The following describes how TLS can be applied to fit a generalised linear model (3) given data $\{\mathbf{w}_n, y_n\}_{n=1}^{\mathcal{N}}$.

In particular, augmenting the feature vector [1] $\boldsymbol{\phi}(\cdot)$ with the targets y, (3) can be re-written

$$\boldsymbol{z}^\top \tilde{\boldsymbol{\omega}} = 0 \tag{8}$$

where $\boldsymbol{z}(\mathbf{x}, y) := (\phi_1(\mathbf{x}), \ldots, \phi_{\mathcal{J}-1}(\mathbf{x}), y)^\top \in \mathbb{R}^{\mathcal{J}}$ and $\tilde{\boldsymbol{\omega}} \in \mathbb{R}^{\mathcal{J}}$ is the vector of parameters.

In this augmented space, instead of minimising the residuals in y (as in (2)), the proposed approach minimises the sum of squared *orthogonal residuals*

$$S_t = \sum_{n=1}^{\mathcal{N}} d_n^2 \tag{9}$$

where d_n is the orthogonal distance from the nth data point to the plane defined by (8)

$$d_n = \boldsymbol{z}_n^\top \hat{\boldsymbol{\omega}} \tag{10}$$

with $\boldsymbol{z}_n := \boldsymbol{z}(\mathbf{w}_n, y_n)$ and $\hat{\boldsymbol{\omega}} := \tilde{\boldsymbol{\omega}}/||\tilde{\boldsymbol{\omega}}||$ (see Fig. 2). It can be shown [18], [19] that the plane minimising (9) must pass through the centroid of the data

$$\bar{\boldsymbol{z}} := \frac{1}{\mathcal{N}} \sum_{n=1}^{\mathcal{N}} \boldsymbol{z}_n. \tag{11}$$

Hence, minimisation of (9) can be achieved by minimising its upper bound

$$\overline{S} = \sum_{n=1}^{\mathcal{N}} ((\boldsymbol{z}_n - \bar{\boldsymbol{z}})^\top \hat{\boldsymbol{\omega}})^2 \tag{12}$$

or in matrix notation

$$\overline{S} = ||\mathbf{Z}^\top \hat{\boldsymbol{\omega}}||^2 \tag{13}$$

where $\mathbf{Z} := ((\boldsymbol{z}_1 - \bar{\boldsymbol{z}})^\top, \cdots, (\boldsymbol{z}_{\mathcal{N}} - \bar{\boldsymbol{z}})^\top)^\top$.

This is equivalent to a low rank matrix approximation problem [20]:

$$\hat{\boldsymbol{\omega}} = \arg\min_{\hat{\mathbf{Z}}} ||\mathbf{Z} - \hat{\mathbf{Z}}||_F, \text{subject to } \text{rank}(\hat{\mathbf{Z}}) < \mathcal{J} - 1, \tag{14}$$

where F is the Frobenious norm.

The total least squares solution is retrieved by finding parameters where $\hat{\mathbf{Z}}^\top \hat{\boldsymbol{\omega}} = 0$, by forming the singular value decomposition of $\mathbf{Z} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$, where $\boldsymbol{\Sigma}$ is a diagonal matrix of the singular values, and $\mathbf{U}$ and $\mathbf{V}$ are matrices of the left and right singular vectors. The solution is retrieved by selecting elements of the right singular vector corresponding to the smallest singular value [21].:

$$\hat{\boldsymbol{\omega}} = \frac{-1}{V_{\mathcal{J},\mathcal{J}}} \mathbf{V}_{(\mathcal{J}_1,\ldots,\mathcal{J}_{\mathcal{J}-1})} \tag{15}$$

The intercept term is then formed as:

$$\hat{\omega}_0 = -\bar{\boldsymbol{z}}^\top \hat{\boldsymbol{\omega}}, \tag{16}$$

[1] Note that, to avoid biasing effects due to the mapping of $\boldsymbol{\varepsilon}$ into the feature space, the feature vector as far as possible should be chosen such that the distribution of $\phi_j(\mathbf{w})$ around $\phi_j(\mathbf{x})$ for $j \in 1, \cdots, \mathcal{J}$ is symmetric. In practice, this condition is not found to be crucial in obtaining a superior fit over approaches that ignore the input errors $\boldsymbol{\varepsilon}$.

and the plane which minimises the squares residuals orthogonal to the predicted curve is given by:

$$\tilde{\boldsymbol{\theta}} = (\hat{\boldsymbol{\omega}}^\top, \hat{\omega}_0)^\top \quad (18)$$

Similarly to the standard approach (§II-C), scalability can be improved by fitting spatially localised low order polynomials. The total least squares algorithm can be modified by the use of a weighting matrix $\boldsymbol{\Lambda}$, and the objective function (9) can be modified into the weighted objective function minimising the weighted sum of squared orthogonal residuals [22]:

$$S_{tw} = \sum_{n=1}^{\mathcal{N}} \lambda_n d_n^2. \quad (18)$$

Then, instead of the plane passing through the centroid of the data (as in (11)), it must now pass through the weighted mean centre [22], denoted as

$$\bar{\boldsymbol{z}} := \frac{\sum_{n=1}^{\mathcal{N}} \lambda_n \boldsymbol{z}_n}{\sum_{n=1}^{\mathcal{N}} \lambda_n}. \quad (19)$$

The data matrix is formed as

$$\mathbf{Z} := ((\sqrt{\lambda_1}(\boldsymbol{z}_1 - \bar{\boldsymbol{z}}))^\top, \cdots, (\sqrt{\lambda_n}(\boldsymbol{z}_{\mathcal{N}} - \bar{\boldsymbol{z}}))^\top)^\top. \quad (20)$$

and the optimal solution retrieved by forming the singular value decomposition of $\mathbf{Z}$, and computing the parameters using equations (17)-(16). Similarly to the weighted least squares implementation in §II-C, if $\boldsymbol{\Lambda} = \mathbf{I}$, the weighted TLS formulation reduces to the global model.

B. Nongeneric TLS

In some problems, the singular value decomposition of the data matrix $\mathbf{Z}$ can fail to produce a finite solution, or unstable results. A solution to this problem is to use an algorithm proposed in [23], whereby $\tilde{\boldsymbol{\omega}}$ is chosen as another right singular vector corresponding to a larger singular value (which is either not zero, or is above an acceptable value determined a priori). In this paper, the implementation of the nongeneric TLS solution selects the optimal singular vector through a cross validation process on the possible singular values.

C. Weighting with Errors-in-variables

In spatially localised models, a weighting function is used to compute the contribution of a sample to a local model's parameter estimate, based on its distance in the input space from the model centre. For this, in ordinary least squares learning, a number of weighting functions have been proposed, such as the tricube or Gaussian functions, or a simply a piecewise discretisation of the space [24]. However, if the inputs are noisy, then some care must be taken into how these functions are chosen due to the difficulty in estimating the distance of samples $\mathbf{w}_n$ from nearby local models.

In particular, if the distance calculation is incorrect, then noisy samples may be allocated to the wrong local model (or over-weighted in the case of smooth weighting functions). In turn, this can affect the contribution of samples to the local model fit, thereby potentially reducing the quality of the overall fit.

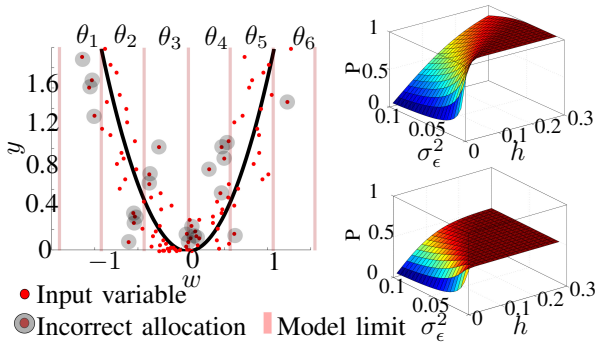

Fig. 3: Incorrect model allocation using a piecewise weighting function, due to additive error on the input. (Left) Samples with noise on the input are denoted by red dots, samples circled are assigned to incorrect models (boundaries of model shown by light red vertical lines). Solid black line denotes true function. (Right) Probability of noisy sample being assigned to the correct model, when varying model bandwidth h and noise standard deviation σ_ε^2, given that (top) $\mathbf{x}_n$ is positioned at the centre of the model, and (bottom) $\mathbf{x}_n$ is positioned at the boundary of the model.

To see this, consider the task of learning a one dimensional function (see Fig. 3 (left)) subject to zero-mean Gaussian noise on the inputs $\varepsilon \sim N(0, \sigma_\varepsilon^2)$, and using a simple piecewise weighting function

$$\lambda_{n,i} = \begin{cases} 0 & \|x_n - c_i\| > h \\ 1 & \|x_n - c_i\| \leq h \end{cases} \quad (21)$$

where c_i is the model centre and h is the bandwidth of the model. In this case, the probability that a given sample w_n is allocated to the correct model is

$$P(a \leq w_n \leq b) = \int_a^b \frac{1}{\sqrt{2\pi\sigma_\varepsilon^2}} N(x_n, \sigma_\varepsilon^2)\, \mathrm{d}x \quad (22)$$

where $a = c_i - \frac{h}{2}$ is the lower limit of the model, and $b = c_i + \frac{h}{2}$ is the upper limit. This equation can be evaluated by taking the difference between the cumulative distribution functions at b and a.

From (22), the probability of correct allocation depends on three factors: (i) the location of the model centre relative to x_n, (ii) the bandwidth of the model h, and (iii) the standard deviation of the input noise σ_ε^2. For instance, in this example, if the true input coincides with the ith model centre (i.e., $x_n = c_i$), and the bandwidth is equal to the noise standard deviation ($h = \sigma_\varepsilon^2$), the probability of w_n being correctly allocated to the model is $P = 0.383$. This probability increases to 1 as the bandwidth size increases (Fig. 3, top). On the other hand, if the true input is located on the boundary between two models ($x_n = c_i \pm h/2$), there is a maximum 0.5 probability of correct allocation (Fig. 3, bottom).

This problem is not limited to the piecewise model, since even with weighting functions such as the Gaussian [25]

$$\lambda_{n,i} = \exp\left(\frac{-(\mathbf{w}_n - \mathbf{c}_i)^2}{2h}\right) \quad (23)$$

noisy samples may still be given excessive weight by incorrect (nearby) models (although, as the latter have weights decreasing asymptotically with distance, this tends to mitigate these problems).

Considering these factors, it would seem that the optimal approach would be to maximise the bandwidth of local models in order to minimise the probability of noisy samples being allocated to the wrong models. However, in the context of learning, excessively large bandwidths may also cause *over-smoothing* and thereby reduced accuracy of predictions. In practice, a trade-off must be made between the problems of model allocation and over-smoothing. In the experiments reported in this paper, this is achieved through use of the Gaussian weighting function (23), with the parameter h selected through cross-validation. The interested reader is referred to [26] for further discussion of the choice of weighting functions for use in nonparametric estimation with errors-in-variables.

D. Motion Prediction from Noisy Sensor Readings

Having learnt the model parameters $\tilde{\boldsymbol{\theta}}$, the next step is to form predictions based on incoming sensor readings.

In standard function approximation (see §II-C), the movement estimate $\tilde{y}^*$ for a given feature vector query point $\boldsymbol{\phi}(\mathbf{w}^*)$ in a global model is simply

$$\tilde{y}^* = \boldsymbol{\phi}^\top \tilde{\boldsymbol{\theta}} \tag{24}$$

while for the spatially localised models, $\tilde{y}^*$ is given as the normalised weighted sum of the predictions from all models [16]

$$\tilde{y}^* = \frac{\sum_{c=1}^{\mathcal{C}} \lambda_c \boldsymbol{\phi}^\top \tilde{\boldsymbol{\theta}}}{\sum_{c=1}^{\mathcal{C}} \lambda_c}. \tag{25}$$

However, because this fails to take account of the noise in $\mathbf{w}^*$, it can result in poor accuracy.

The ideal prediction would be obtained by directly feeding $\mathbf{x}^*$ to the model (1), but in the present context this reading is not directly accessible. It is therefore necessary to build an estimate $\tilde{\mathbf{x}}^*$ based on the data available.

In this paper, this is achieved through use of *replicate data* [17], whereby for any query point $\mathbf{x}^*$, the availability of $\mathcal{K}$ noisy replicates

$$\mathbf{w}_k = \mathbf{x}^* + \boldsymbol{\varepsilon}_k \tag{26}$$

is assumed within the test data. Under the assumption of zero-mean distribution of errors $\boldsymbol{\varepsilon}_k$ (7), this means that a simple estimate $\mathbf{x}^*$ can be obtained by taking the sample mean of the replicates over $\mathcal{K}$. This allows the final prediction to be made by using the feature vector $\boldsymbol{\phi}(\tilde{\mathbf{x}}^*)$ with equation (24) for global models, or equation (25) for local models.

Note that, in general, the accuracy of the prediction $\tilde{\mathbf{x}}^*$ depends on the number of replicate data available at that point. For sensors measuring continuous variables (as considered here), exact replicates $\mathbf{w}_k$ of sensor readings at a given $\mathbf{x}^*$ are unlikely to be available. However, in practice, a good estimate can still be found from approximate replicates (i.e., using samples $\mathbf{w}_n = \mathbf{x}_n + \boldsymbol{\varepsilon}_n$ where $||\mathbf{x}_n - \mathbf{x}^*||$ is small).

In the experiments reported here, approximate replicates are obtained by a heuristic binning procedure. In this, a $\mathcal{P}$-dimensional grid of bins is created, each with a fixed width. The training inputs are placed in bins according to their value. The query point $\mathbf{w}^*$ is placed in a bin, and the mean of that bin is used as the estimate $\tilde{\mathbf{x}}^*$.

IV. Evaluation

In this section, the proposed approach is evaluated through a simulation study, and through an experiment on acceleration data from a fabric-embedded device.

A. Simulation

The goal of the first evaluation is to characterise the performance of the proposed approach for learning and predicting movements from noisy sensory inputs. For this, learning is tested on artificial data from models with both linear and non-linear relationships between the input $\mathbf{x}$, the sensed $\mathbf{w}$, and the target quantity y. For example, $\mathbf{x}$ may represent the acceleration of a body segment (e.g., forearm), $\mathbf{w}$ the fabric sensor readings (e.g., from a shirt sleeve), and y the corresponding acceleration of another segment (e.g., hand), see Fig. 1.

1) 1-Dimensional Input: In this evaluation, a set of $\mathcal{N}$ points for training the model is generated as follows. As inputs to the model, first, a set of $\mathcal{M} = 50$ independent sample inputs are drawn from the uniform random distribution $x_m \sim U[-1, 1]$. Each of these is then corrupted with additive Gaussian noise. To simulate multiple sensor readings observed from the same true input, but with different noise corruptions at each sampling, this process is repeated $\mathcal{K} = 10$ times, to generate the matrix of data $\mathbf{W} \in \mathbb{R}^{\mathcal{M}\times\mathcal{K}}$, where each column of the matrix is a corruption of the true input,

$$w_{m,k} = x_m + \varepsilon_k \tag{27}$$

where $\varepsilon_k \sim N(0, \sigma_\varepsilon^2)$ and $\sigma_\varepsilon^2 = 0.15$.

The matrix of data is then transformed[2] to the vector of data $\mathbf{w} = vec(\mathbf{W})^T \in \mathbb{R}^{\mathcal{M}\mathcal{K}}$, where $\mathcal{M}\mathcal{K} = \mathcal{N}$.

At the same time, the corresponding target quantities y_n are computed for each of the readings w_n

$$y_n = y_{m,k} = f(x_m) + \epsilon_n \tag{28}$$

where $\epsilon_n \sim N(0, \sigma_\epsilon^2)$ and $\sigma_\epsilon^2 = 0.01$. In the following, results are reported for generative functions f that are (i) linear $f_1(x) = 1.5x + 3$, (ii) quadratic $f_2(x) = 4x^2 + 0.75x + 3$, and (iii) sinusoidal $f_3(x) = -0.3\sin(2.5x)$ in the inputs.

The resultant $\{w_n, y_n\}_{n=1}^{\mathcal{N}}$ are used to train the approximator (3) through the total least squares (TLS) method outlined in §III. For this, $\boldsymbol{\phi}$ is chosen according to the model (f_1, f_2 and f_3) used to generate the data. In particular, for f_1 and f_2 basis functions exactly capturing the parametric form of the model (e.g., for f_2, $\boldsymbol{\phi}(x) := (x^2, x, 1)^\top$) are used, while for f_3, a 3rd order polynomial basis is used to estimate the function. This is to test the case of underlying function being

[2]For further details on the vec transformation, the reader is directed to "Petersen and Pedersen, "The Matrix Cookbook", 2012"

		Linear	Quadratic	Sinusoidal
LS	$\|\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|\|$	0.467 ± 0.053	3.188 ± 0.164	-
	NMSE	0.344 ± 0.043	0.736 ± 0.081	0.493 ± 0.041
TLS	$\|\|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}\|\|$	0.189 ± 0.068	1.234 ± 0.368	-
	NMSE	0.067 ± 0.013	0.370 ± 0.136	0.265 ± 0.052

TABLE I: Mean norm difference between estimated and ground truth parameters and normalised mean squared error (NMSE) in predictions $\tilde{y}^*$. Results are mean±s.d. over 20 trials.

unknown. For comparison, identical models are trained on the same data through standard least squares (LS), using the approach outlined in §II-C [3]. The procedure is repeated for 20 trials on different data sets.

Table I summarises the results. There, it can be seen that the parameters estimated by TLS for f_1 and f_2 are are much closer to the ground truth, as compared to those learnt through standard least squares. This is reflected in the normalised mean squared error values (NMSE), that indicate good predictive accuracy of the models. Likewise, for f_3, TLS obtains lower NMSE than LS, despite the exact parametric form of the function being unavailable in this case.

In order to evaluate the effectiveness of the estimation method for the predictors (§III-D), the mean squared error (MSE) of the predictor residuals was computed. In the naive LS case, this was the residual of the noisy input $\mathbf{w}^*$ against the true input $\mathbf{x}^*$ and in the TLS case the residual between the estimated true input $\tilde{\mathbf{x}}^*$ and the true input. The mean results obtained from the linear data set were 0.146 ± 0.012 for the $\mathbf{w}^*$ MSE, and 0.015 ± 0.002 $\tilde{\mathbf{x}}^*$ MSE. Similar results were seen for the other functions in this experiment, as this procedure estimated the residuals independently of the prediction y^*. It can be seen from these results that the heuristic binning method outlined in §III-D computes predictors $\tilde{\mathbf{x}}^*$ that are closer to the ground truth.

Interestingly, the worst performance for both methods is found when learning the quadratic function f_2. This appears to be due to the specific form of this function, where high input noise tends to cause overlap of data between the two 'arms' of the parabola, resulting in interference in learning (shown in Fig. 4).

These results are verified by examining the prediction curves of the learnt models over the range of training data. In Fig. 4, the predictions of the learnt TLS models for f_2 and f_3 are plotted, overlaid upon the ground truth values, and those of standard LS[4]. It can be seen that the models learnt with TLS are in good agreement with the underlying ground truth functions. In contrast, those learnt through standard LS suffer

[3]Note that the noiseless prediction method (III-D) is used only for models trained with TLS. The LS procedure does not make use of prior knowledge about noise on the inputs either in the training (II-C) or prediction stages.

[4]Note that, for standard LS, the predictions extend over a wider range of inputs since the $w_n = x_n + \varepsilon_n$ usually extends beyond the maximum and minimum x_n due to the symmetrically distributed additive noise.

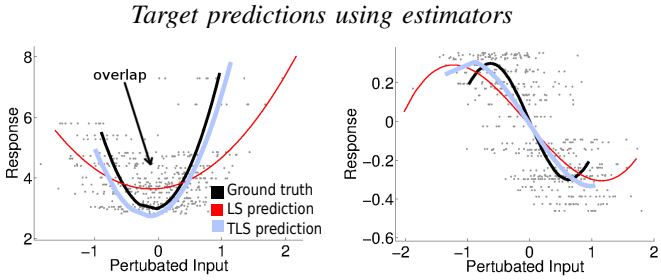

Fig. 4: Least squares (thin red) and total least squares (thick light purple) predictions overlaid on the ground truth (black) for f_2 (left), and f_3 (right) when learning on noisy data (grey dots).

a bias towards zero, causing attenuation of the predictions and thereby higher errors.

To further assess the performance of learning, the experiment was repeated, varying (i) the noise in the data, $0 \leq \sigma_\varepsilon^2 \leq 0.4$, and (ii) the number of replicates available, $1 \leq \mathcal{K} \leq 30$. Note that, the former corresponds to increasing the 'slack' of the fabric-mounted sensor (since looser coupling between sensor and wearer is likely to result in larger motion artefacts), while the latter corresponds to differences in the size and density of the data set recorded during the calibration stage (ref. §II-B).

The results for functions f_1, f_2 and f_3 are plotted in Fig. 5. There, it can be seen that, as expected, there is a decrease in accuracy for both TLS and LS as the noise level increases. However, the divergence of the TLS and LS lines indicates a much quicker degradation of performance for the latter. For the concave function f_2, this is particularly pronounced, an effect that may also be attributed to the non-monotonicity of the function: increasing noise causes greater overlap of data from the two arms of the parabola resulting in greater interference during learning.

Looking at the learning curves for varying $\mathcal{K}$ (Fig. 5, right), it can be seen that the error in the prediction NMSE for TLS drops rapidly as the number of replicates found in the data increases, levelling off at around $\mathcal{K} = 10$ for all functions. This suggests that the proposed approach is able to use the data efficiently to obtain a good fit. The LS line, in contrast, does not change significantly, despite the increase in the amount of data available.

2) Multi-Dimensional Input: In this section, the learning methods outlined in §III are evaluated in a multi-dimensional setting. Higher dimensional problems are common in many applications, for example in the context of analysing human movement the input $\mathbf{x}$ may represent both the pitch and roll of the arm, from which the corresponding end-effector position is to be predicted.

Here, learning is evaluated on two nonlinear functions, namely,

$$f_1(x_1, x_2) = x_2^2 + 0.5x_2 + 1.5x_1^2 + 0.75x_1 + 3 \qquad (29)$$

and

$$f_2(x_1, x_2) = x_1^2 - x_2^2 + \arctan(3x_1 + 4x_2). \qquad (30)$$

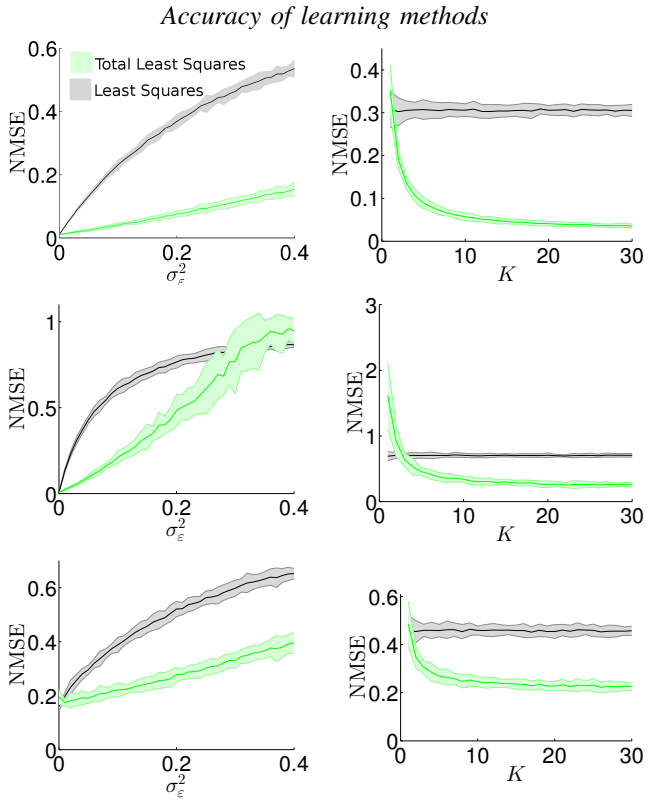

Fig. 5: Effects of varying σ_ε^2 (left) and $\mathcal{K}$ (right) on NMSE for linear (top), quadratic (middle) and sinusoidal (bottom) functions when learning with least squares (black) and total least squares (light green). Shown mean± two s.d. over 20 trials.

The procedure is as follows. A set of $\mathcal{M} = 50$ training samples are drawn from the uniform random distribution $x_i \sim U[-1, 1]$, where $i \in \{1, 2\}$. For each of these, to simulate noise and motion artefacts, a set of $\mathcal{K} = 10$ readings are generated, corrupted with additive Gaussian noise

$$\mathbf{w}_n = \mathbf{w}_{m,k} = \mathbf{x}_m + \boldsymbol{\varepsilon}_k \tag{31}$$

where $\boldsymbol{\varepsilon}_k \sim N(\mathbf{0}, \sigma_\varepsilon^2 \mathbf{I})$ and $\sigma_\varepsilon^2 = 0.05$. At the same time, the corresponding target quantities y_n are computed for each of the readings $\mathbf{w}_n$

$$y_n = y_{m,k} = f(\mathbf{x}_m) + \epsilon_n \tag{32}$$

where $\epsilon_n \sim N(0, \sigma_\epsilon^2)$ and $\sigma_\epsilon^2 = 0.001$.

As in §IV-A1, the resultant $\{\mathbf{w}_n, y_n\}_{n=1}^{\mathcal{N}}$ are used to train the approximator (3) through both total least squares (TLS) (§III), and the standard least squares (LS) outlined in §II-C. In this higher dimensional problem, local linear models are used to estimate the functional relationship due to their improved scalability in terms of data dimensionality. Local models are placed in an equally spaced 10×10 grid, covering the range of $\mathbf{w}$, with the width of each model selected a priori as $h = 0.05$ by a visual observation of the fit. The procedure is repeated for 20 trials on different data sets. Where necessary, the nongeneric TLS algorithm (§III-B) is used to prevent numerical errors.

		Quadratic	Atan
LS	$\tilde{y}^*$ NMSE	0.321 ± 0.078	0.249±0.058
TLS	$\tilde{y}^*$ NMSE	0.191± 0.079	0.143±0.058

TABLE II: Normalised mean squared error (NMSE) in predictions $\tilde{y}^*$, and normalised mean squared error in the predictors, for given functions estimated local linear models. Results are mean±s.d. over 20 trials.

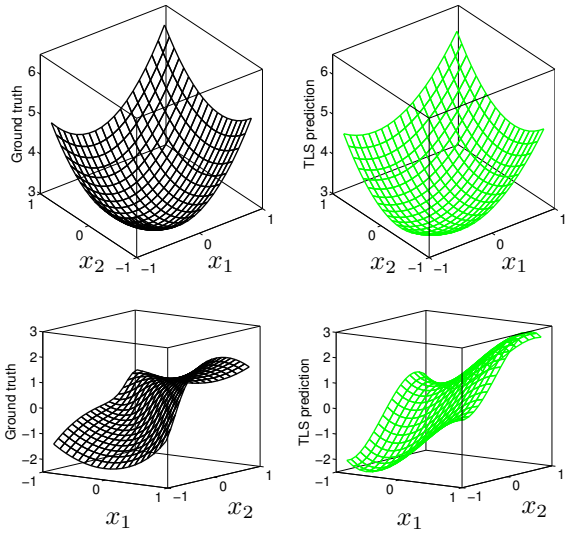

Fig. 6: Functions f_1 (top) and f_2 (bottom), showing estimations using local linear TLS (right).

Table II summarises the results for the 2-dimensional input, and Fig. 6 shows the surfaces estimated for f_1 and f_2, compared against the true surfaces. The normalised mean squared error (NMSE) values of the prediction are lower when using the local TLS models for both functions. This is consistent with the results shown in §IV-A1, even when estimating the function non-parametrically in a higher-dimensional space. For comparison, fitting a global quadratic model of f_1 with $\boldsymbol{\phi}(\mathbf{w}) := (w_1^2, w_2^2, w_1, w_2, 1)^\top$, the average NMSE value using LS is 0.328 ± 0.051 and using TLS is 0.077 ± 0.0027, indicating that the local TLS approach outperforms LS, even when the latter is provided with a priori information such as the correct parametric form of the model.

To examine the trade-off between model allocation and over-smoothing (ref. §III-C), the experiment for learning f_1 was also repeated while varying the model bandwidth h. Fig. 7 shows the mean NMSE of the local TLS model for different levels of input noise $\sigma_\varepsilon^2 \in \{0, 0.02, 0.05, 0.1\}$ for $0.05 \leq h \leq 0.5$. As can be seen, for $\sigma_\varepsilon^2 = 0$, the optimal bandwidth with respect to minimising the NMSE is small, due to the local parameter estimate being a good fit from the local data. As the bandwidth increases, the NMSE begins to rise, as the estimated function becomes oversmoothed. At larger σ_ε^2, however, the optimal bandwidth h increases (vertical black

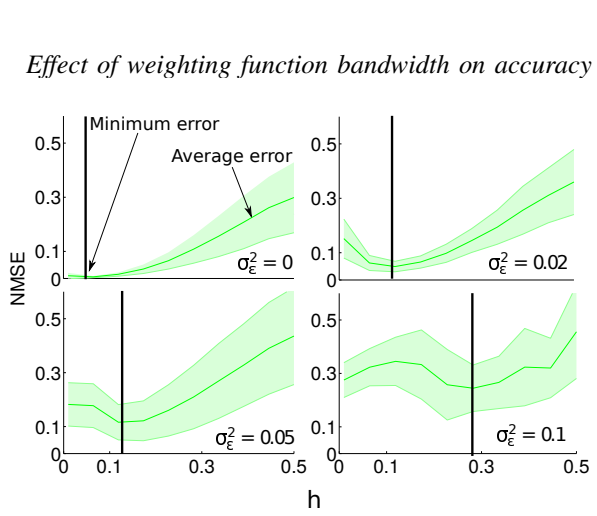

Fig. 7: Effects of varying bandwidth (h) of weighting function on NMSE for f_1, when learning with total least squares, while varying amounts of measurement variance σ_ε^2. Black bar shows minimum NMSE value. Shown mean $\pm$ s.d. over 20 trials.

lines). This is consistent with the discussion in §III-C, which showed that models using smaller bandwidths would underperform in the presence of measurement error, due to he problem of samples being wrongly allocated to nearby local models.

B. Experiment - Fabric Mounted Sensor

In this evaluation, the proposed approach is tested for learning from physical data with the goal of predicting the movement of an object through space from a fabric-mounted sensor. The experimental platform used is shown in Fig. 8.

The platform consists of a pair of LilyPad ADXL335 tri-axial accelerometers mounted onto a plastic box. Of these, the first is attached rigidly, to provide a ground truth measurement of the box acceleration α, while the second is sewn onto a light-weight strip of cloth, and measures the fabric acceleration β. The cloth attachment is designed such that the slackness of the fabric s (defined as the maximum displacement from the box admitted by the fabric, see Fig. 8) can be adjusted between $s = 0\,cm$ (taut against the box) and $s = 6\,cm$.

During motion, readings from the two sensors are sampled synchronously at a rate of $23\,Hz$ using an Arduino Uno (Atmega-328P microcontroller, 16-bit ADC), and sent wirelessly to a PC base-station for analysis. The fabric-mounted sensor is connected to the Arduino using conductive thread to ensure minimal interference with the fabric motion (as might occur, for example, with use of wiring). Note that, while this reduces invasiveness of the sensor, it also adds further noise to the sensor readings [27], making the learning task in this experiment especially challenging.

As data, signals from the two sensors recorded during random shaking of the box for sessions of $60\,s$ each are used. The raw signals are preprocessed by converting the ADC

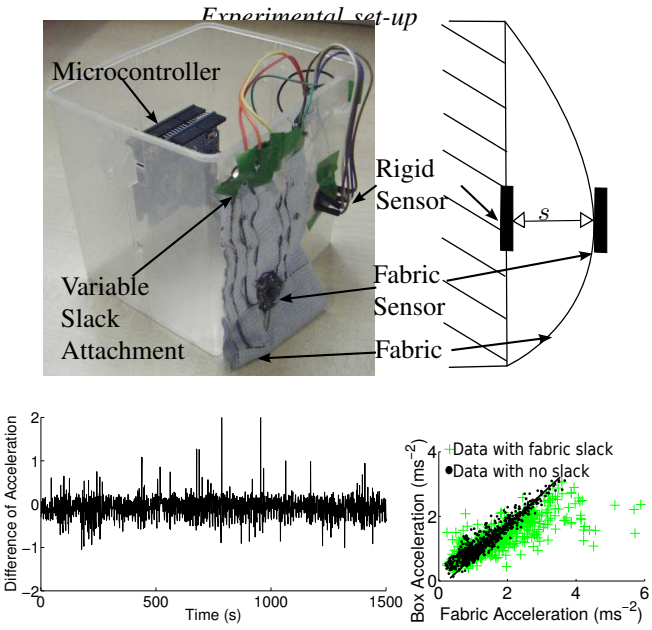

Fig. 8: Experimental set up (top). Difference of fabric and box acceleration magnitude against time (bottom left). Plot of box acceleration magnitude against that of the box (bottom right) showing ground truth mapping (black line), data with no slack (black circles) and data with slack (crosses).

values to acceleration in g, and then computing the acceleration magnitude for each time step. The later is commonly used in clinical movement recording studies [28] and is used in this experiment to verify the usefulness of the proposed approach in such settings. The resultant data $\{\alpha_n, \beta_n\}_{n=1}^{\mathcal{N}}$ consists of $\mathcal{N} = 1350$ samples of box- and fabric-mounted sensor readings, respectively. These are randomly split into training and test data sets of equal size.

An sample data set is illustrated in Fig. 8 (bottom left and right), for displacement $s = 6\,cm$. On the left, the difference between fabric and box acceleration magnitude is shown against time, showing a significant amount of noise, much greater than would be expected from ordinary electrical noise. On the right, the fabric acceleration magnitude is plotted against that of the box, for two different slacks. In this case, the sensors are calibrated against one another so that there exists an identity relationship between the two (black line). When the slack is zero ($s = 0\,cm$, black dots) the sensor readings lie closely along the line, however for greater slack ($s = 6\,cm$, green crosses) a much broader distribution of readings is observed.

For learning, the proposed TLS approach is then used to train a linear model $\boldsymbol{\phi} := (x, 1)^T$ mapping the measured fabric accelerations β to box acceleration α on the training data. Note that, at the prediction stage, exact replicates of the form (26) are not available for forming the estimate ε_n (ref. §III-D). Instead, approximate replicates are obtained by grouping similar values of β_n together into 350 discrete bins of equal size and spacing, and treating the data in these bins as the replicates.

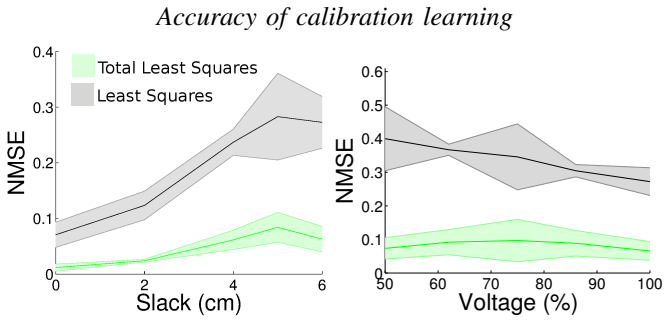

Fig. 9: NMSE of least squares (black) and total least squares (light green) of box motion when varying fabric slacknesses (left) and fabric sensor voltage (right). Results are mean±s.d. over 5 trials.

In the following, results are reported for 5 trials of this experiment, in which data were recorded at slackness levels $s = 0$, 2, 4, 5, and 6 cm. As a baseline for comparison, the experiment is also repeated using the same model but learnt through the standard LS approach (ref. §II-C). Predictions are made offline (i.e. not during motion), as is consistent with rehabilitation studies which monitor readings over a long-term period.

In an initial experiment, the identity function was learnt between the two calibrated accelerometers. These known parameters were used to verify that the estimated parameters were correct. These results are shown in Fig. 9 (left) where the mean prediction NMSE of the LS and TLS models are plotted against the slackness s. There it can be seen that, as the slackness increases, there is a gradual decrease in accuracy for both methods. This is in agreement with the simulation study (see §IV-A), where it was seen that increasing noise (motion artefacts) resulted in a similar trend. However, comparing TLS with the standard LS approach, it is seen that the proposed approach consistently outperforms the latter across the range of slacknesses, with a more gradual decrease in accuracy in the face of greater noise.

To further test performance, the experiment was also repeated using data from a decalibrated pair of sensors. This corresponds to the situation described in §II-B, where a temporary (potentially heterogeneous) sensor is used for gathering data in a calibration stage.

In this experiment, the same test platform in Fig. 8 is used and data is collected following the same experimental procedure (with the slack of the fabric fixed at $s = 6\,cm$). However, to induce differences between the two accelerometers, the fabric sensor's input voltage is altered with a potentiometer, reducing the strength of the signal[5]. In the following, results are reported for data collected when the fabric sensor was supplied with (i) 4.3V (100% normal operating voltage), (ii) 3.7V (86%), (iii) 3.2V (75%), (iv) 2.7V (62%), and (v) 2.2V (50%). Note that, decalibrating the sensors in this way induces a non-identity mapping between the sensors, so that, for example, when the fabric sensor operates at 2.15V, the

[5]Note that, at each voltage, the sensors still have each axis calibrated at zero acceleration.

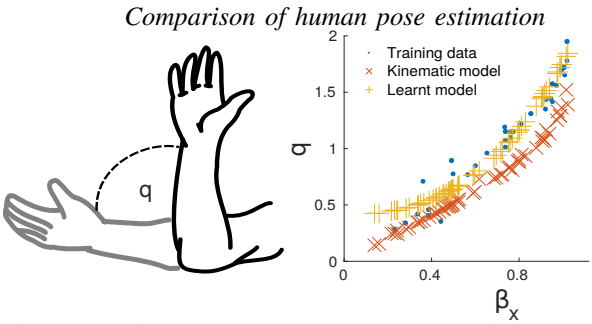

Fig. 10: (Left) Arm segment measurement, (right) predictions of segment angle

readings of the box sensor readings should be approximately twice the magnitude of those of the fabric sensor.

In Fig. 9 (right), the prediction NMSE of the test data against the fabric sensor input voltage are shown for 5 trials of this experiment. As can be seen, the proposed approach outperforms standard LS across the range of voltages (sensor calibration factors).

C. Pose estimation

The proposed learning approach is then evaluated in a human motion context, by predicting the pose of an arm from a fabric embedded device. Pose estimation is particularly important in long-term rehabilitation studies, as it provides quantitative data on a patient's movement, which can be used to monitor progress. For this experiment, the orientation of a forearm is estimated using a sleeve (Fig. 1), with an accelerometer embedded on the inner forearm of the sleeve. The sleeve is loose fitting, to induce motion artefacts.

The experimental setup is illustrated in Fig. 10 (left). In this, the upper arm is placed flat on a surface, while the forearm segment is positioned at different angles around the elbow joint. Recordings were taken using the same accelerometer and Arduino as shown in Sec. IV-B, and a Hall rotary encoder (Melexis, MLX90316) is attached to the elbow, to record the ground truth of the forearm segment orientation.

In this evaluation, predictions made using orthogonal regression techniques outlined in Sec. III are compared against the ground truth values from the encoder. As well as this, comparison predictions are made using the standard regression setup (Sec. II-C), and a kinematic model of the orientation of the arm segment.

This kinematic model of the arm orientation at rest is defined as:

$$q = tan^{-1}(\frac{\sqrt{1-\beta_x^2}}{\beta_x}), \tag{33}$$

where the subscripts denote the x-axis and z-axis of the tri-axel acceleration vector β. This is derived from basic trigonometry [29], and assumes the only acceleration recorded is due to gravity.

To evaluated the proposed method, the recorded pose data is first split randomly into independent training and testing sets of equal size. The standard and orthogonal non-parametric regression models are then learnt, using the x-axis acceleration as input data and the angle obtained from the calibrated Hall

encoder as the predictor. Predictions are made on the testing set of poses. This procedure is repeated 10 times.

These results for one trial are shown in Fig. 10. Here it is shown that the predictions made with the learnt model outperform the kinematic model, even given the bias of being provided with a mathematical model of the mapping. This is also shown in the mean NMSE values of predictions, which for the naive model is 0.358 ± 0.044, the least squares model 0.079 ± 0.070, and total least squares model 0.073 ± 0.025.

V. Discussion

In this paper, the application of statistical learning techniques in estimating motion through minimally invasive, e-textile sensing has been investigated. A crucial difficulty in this area, is the ability to deal with unpredictable motion artefacts introduced into sensing by the complex, unpredictable dynamics of the sensor fabric. The unsuitability of analytic modelling of the latter, both from the view point of model identification and computational tractability, suggests the use of learning approaches. It is also seen, however, that even non-parametric learning techniques need to be adapted to this problem, due to the mismatches in noise assumptions found in standard learning models.

With a view to this, it has been proposed to exploit statistical methods from the errors-in-variables field to deal with the effects of stochastic perturbations to the sensory inputs in clothing-based sensing. An approach to model estimation and movement prediction has been presented, based on the use of total least squares regression for fitting, and estimation of query point inputs during prediction. Evaluation of the proposed approach in simulation has shown its ability to outperform standard regression in fitting non-linear models in face of significant input noise, and to efficiently make best use of replicate readings found in data. Experiments in learning from a physical cloth-mounted accelerometer to estimate the motion of an object in space, and compensate for calibration mismatches, show that it is always an improvement to apply orthogonal regression techniques over ordinary least squares. The computational efficiency of the proposed approach also makes it feasible for implementation onto an embedded device, making it an appealing option for the long-term gathering of data, e.g., in rehabilitation studies.

In future work, the performance of learning from higher dimensional data will be investigated, in order that more complex movements, such as walking, can be captured using sensors integrated into everyday items of clothing. Capturing movements with multiple sensors (e.g. fabric embedded gyroscopes) would also allow for estimating data such as link position or joint angles, which would be of interest to clinicians in diagnostics. An online prediction system will also be investigated to allow for predictions to be made during motion, as opposed to the heuristic binning approach used in this paper.

Acknowledgements

The authors would like to thank Rahul Pathak for building the experimental platform and collecting data, and Karina Thompson for creating the prototype sleeve in Fig. 1.

References

[1] E. Jovanov *et al.*, "A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation," *J. Neuroeng. & Rehab.*, vol. 2, no. 1, p. 6, 2005.

[2] J. Barth *et al.*, "Biometric and mobile gait analysis for early diagnosis and therapy monitoring in parkinson's disease," in *IEEE Int. C. Eng. in Med. & Bio. Soc.*, 2011, pp. 868–871.

[3] T. Shiratori and J. K. Hodgins, "Accelerometer-based user interfaces for the control of a physically simulated character," in *ACM Trans. Graphics*, vol. 27, no. 5. ACM, 2008, p. 123.

[4] D. Roetenberg *et al.*, "Xsens MVN: full 6dof human motion tracking using miniature inertial sensors," Xsens Motion Technologies, Tech. Rep., 2009.

[5] J. Chen *et al.*, "Wearable sensors for reliable fall detection," in *IEEE Int. C. Eng. in Med. & Bio. Soc.* IEEE, 2006, pp. 3551–3554.

[6] T. Liu *et al.*, "Triaxial joint moment estimation using a wearable three-dimensional gait analysis system," *Measurement*, vol. 47, pp. 125–129, 2014.

[7] M. Schepers *et al.*, "Ambulatory estimation of foot placement during walking using inertial sensors," *J. Biomechanics*, vol. 43, no. 16, pp. 3138–3143, 2010.

[8] J. Cancela *et al.*, "A telehealth system for parkinson's disease remote monitoring. the perform approach," in *IEEE Int. C. Eng. in Med. & Bio. Soc.*, 2013.

[9] R. Slyper and J. K. Hodgins, "Action capture with accelerometers," in *ACM SIGGRAPH/Eurographics Symp. Computer Animation*, 2008, pp. 193–199.

[10] S. Min and M. Tianlu, "Cloth animation based on particle model with constraint," in *W.S. Digital Media & Content Man.*, 2011, pp. 141–145.

[11] H. Li and Y. Wan, "An object-oriented system for dynamics-based 3d cloth simulation," in *IEEE Int. C. Inf. Sci. & Tech.*, 2012, pp. 539–545.

[12] B. Chen and M. Govindaraj, "A physically based model of fabric drape using flexible shell theory," *Textile Res. J.*, vol. 65, no. 6, pp. 324–330, 1995.

[13] S. Vijayakumar *et al.*, "Incremental online learning in high dimensions," *Neural Computation*, vol. 17, no. 12, pp. 2602–2634, 2005.

[14] T. Hastie and C. Loader, "Local regression: Automatic kernel carpentry," *Stat. Science.*, pp. 120–129, 1993.

[15] M. Howard and Y. Nakamura, "Locally weighted least squares temporal difference learning," in *Euro. Symp. Art. Neural Networks*, 2013.

[16] S. Schaal and C. G. Atkeson, "Receptive field weighted regression," *ATR Human Information Processing Laboratories, Tech. Rep. TR-H-209*, 1997.

[17] R. Carroll *et al.*, *Measurement Error in Nonlinear Models: A Modern Perspective*. Chapman and Hall, 2006.

[18] Y. Nievergelt, "Total least squares, state of the art in numerical analysis," *SIAM Review*, vol. 36, no. 2, pp. 258–264, 1994.

[19] G. H. Golub and C. F. Van Loan, "An analysis of the total least squares problem," *SIAM J. Num. Analysis*, vol. 17, no. 6, pp. 883–893, 1980.

[20] I. Markovsky and S. Van Huffel, "Overview of total least-squares methods," *Signal processing*, vol. 87, no. 10, pp. 2283–2302, 2007.

[21] W. Gander, *The Singular Value Decomposition*, Swiss Federal Institute of Technology in Zurich, 2008.

[22] C. M. Shakarji and V. Srinivasan, "Theory and algorithms for weighted total least-squares fitting of lines, planes, and parallel planes to support tolerancing standards," *J. Comp. & Inf. Sci. in Eng.*, vol. 13, no. 3, p. 031008, 2013, weighted Total Least Squares.

[23] S. Van Huffel and J. Vandewalle, "Analysis and solution of the non-generic total least squares problem," *SIAM J. on Matrix Analysis. and Apps.*, vol. 9, no. 3, pp. 360–372, 1988.

[24] W. S. Cleveland and S. J. Devlin, "Locally weighted regression: An approach to regression analysis by local fitting," *J. American Stat. Assoc.*, vol. 83, no. 403, pp. 596–610, 1988.

[25] S. S. Keerthi and C.-J. Lin, "Asymptotic behaviors of support vector machines with gaussian kernel," *Neural Computation*, vol. 15, no. 7, pp. 1667–1689, 2003.

[26] J. Fan and Y. K. Truong, "Nonparametric regression with errors in variables," *Annals of Stat.*, pp. 1900–1925, 1993.

[27] T. Linz *et al.*, "Embroidering electrical interconnects with conductive yarn for the integration of flexible electronic modules into fabric," in *IEEE Int. Symp. Wearable Computers*. IEEE, 2005, pp. 86–89.

[28] M. Albert *et al.*, "Fall classification by machine learning using mobile phones," *PLoS ONE*, vol. 7, no. 5, p. e36556, 2012.

[29] K. Tuck, "Tilt sensing using linear accelerometers," *Freescale Semiconductor Application Note AN3107*, 2007.